TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI TP. HỒ CHÍ MINH

KHOA CÔNG NGHỆ THÔNG TIN


# BÁO CÁO THỰC HÀNH

## MÔN HỌC: KHAI THÁC DỮ LIỆU

## NHÓM : 2+12

## ĐỀ TÀI: XÂY DỰNG MÔ HÌNH DỰ ĐOÁN CUSTOMER CHURN THEO QUY TRÌNH CRISP-DM

**Thành viên nhóm:**


| MSSV | Họ và tên | Lớp |
|---|---|---|
| 2251120023 | Trần Trung Kiên | CN22A |
| 2251120043 | Nguyễn Minh Quân | CN22A |
| 2251120314 | Nguyễn Trọng Thái | CN22F |
| 2251150002 | Trần Đặng Đình Cơ | KM22A |
| 2251150019 | Nguyễn Hữu Kiều (NT) | KM22A |
| 2251150012 | Võ Trần Hoàng | KM22A |
| 096205010694 | Lý Duy Khang | CN2301A |


TP Hồ Chí Minh, tháng 11 năm 2025

**Tóm tắt**

"Báo cáo này trình bày quy trình xây dựng mô hình dự đoán khách hàng rời bỏ (Customer Churn) dựa trên bộ dữ liệu viễn thông. Nhóm thực hiện theo chuẩn CRISP-DM, bắt đầu từ việc phân tích dữ liệu (EDA), xử lý missing value và tạo đặc trưng mới. Sau quá trình thực nghiệm và so sánh các thuật toán, kết quả cho thấy mô hình **Logistic Regression** hoạt động hiệu quả nhất với độ ổn định cao và khả năng nhận diện đúng khách hàng rủi ro (Recall) vượt trội hơn các mô hình phức tạp. Cuối cùng, mô hình này được lựa chọn để đóng gói và triển khai lên ứng dụng web demo nhằm hỗ trợ bộ phận kinh doanh."

# 1. BUSINESS UNDERSTANDING (HIỂU NGHIỆP VỤ)

## 1.1. Bối cảnh và vấn đề kinh doanh

Trong bối cảnh cạnh tranh khốc liệt của ngành viễn thông, các doanh nghiệp phải đối mặt với bài toán **khách hàng rời bỏ dịch vụ (Customer Churn)**. Khi khách hàng ngừng sử dụng dịch vụ, doanh nghiệp không chỉ mất doanh thu trực tiếp mà còn phải tốn chi phí gấp 5–7 lần để thu hút một khách hàng mới so với việc duy trì khách hàng cũ.

Việc tận dụng dữ liệu lịch sử về hành vi sử dụng, hóa đơn, và loại hợp đồng cho phép doanh nghiệp xây dựng các mô hình dự đoán sớm. Mục tiêu là xác định đúng người, đúng thời điểm để triển khai các chương trình chăm sóc và ưu đãi giữ chân.

## 1.2. Mục tiêu dự án

Dự án được thực hiện với hai nhóm mục tiêu chính:

- **Mục tiêu Kinh doanh (Business Objective):** Giảm tỷ lệ rời bỏ, tối ưu hóa chiến dịch chăm sóc khách hàng (Retention Campaign), từ đó tăng doanh thu dài hạn và cải thiện sự hài lòng của người dùng.
- **Mục tiêu Khai phá dữ liệu (Data Mining Objective):** Xây dựng mô hình phân loại nhị phân (Binary Classification) để dự đoán nhãn Churn (Yes/No) với độ chính xác cao nhất có thể trên tập dữ liệu kiểm thử.

## 1.3. Chỉ số đánh giá (KPI)

Nhóm xác định các thước đo hiệu quả mô hình bao gồm:

- **Accuracy:** Độ chính xác tổng thể.
- **Recall (Quan trọng nhất):** Tỷ lệ phát hiện đúng khách hàng rời bỏ. Trong bài toán này, việc bỏ sót khách hàng sắp rời đi (False Negative) gây thiệt hại lớn hơn là dự đoán nhầm, do đó Recall được ưu tiên tối ưu.
- **F1-Score:** Trung bình điều hòa giữa Precision và Recall, đặc biệt quan trọng khi dữ liệu bị mất cân bằng.
- **ROC-AUC:** Khả năng phân loại của mô hình ở các ngưỡng khác nhau.

# 2. DATA UNDERSTANDING (HIỂU DỮ LIỆU)

## 2.1. Mô tả tập dữ liệu

Dataset sử dụng trong dự án: **Customer_Churn.csv**
Dữ liệu đầu vào bao gồm các nhóm thông tin chính của khách hàng:

- **Nhân khẩu học:** Giới tính (Gender), Người cao tuổi (SeniorCitizen), Tình trạng hôn nhân (Partner), Người phụ thuộc (Dependents).
- **Thông tin tài khoản:** Thời gian gắn bó (Tenure), Loại hợp đồng (Contract), Phương thức thanh toán (PaymentMethod), Hóa đơn điện tử (PaperlessBilling).
- **Dịch vụ sử dụng:** Dịch vụ điện thoại, Internet (DSL/Fiber Optic), Bảo mật trực tuyến, Hỗ trợ kỹ thuật, Streaming TV/Movies.
- **Thông tin chi phí:** Chi phí hàng tháng (MonthlyCharges) và Tổng chi phí tích lũy (TotalCharges).
- **Biến mục tiêu (Target):** Churn (Yes - Rời bỏ / No - Ở lại).

## 2.2. Đánh giá chất lượng dữ liệu

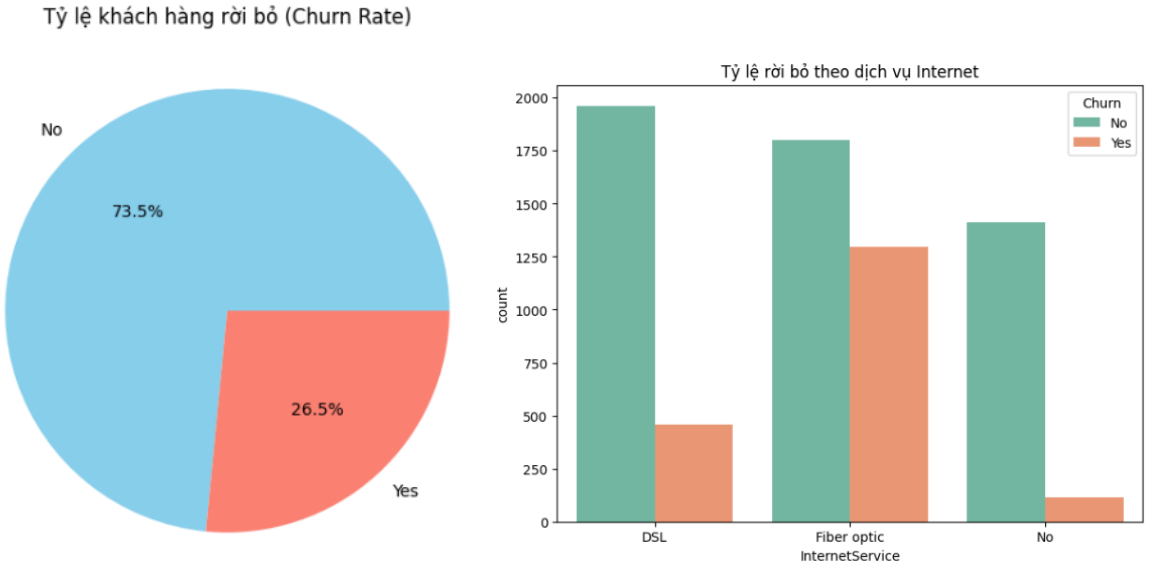
Qua quá trình kiểm tra sơ bộ, nhóm phát hiện các vấn đề sau:

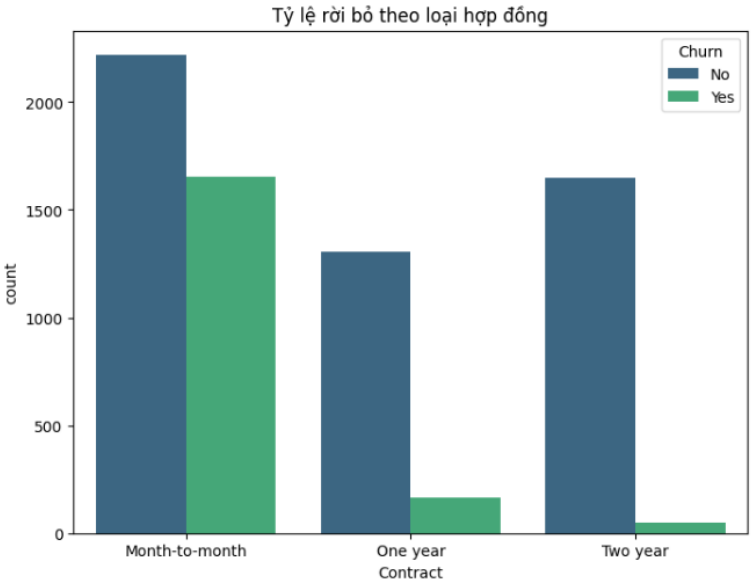
| Vấn đề | Chi tiết | Hướng xử lý đề xuất |
|---|---|---|
| **Missing Values** | Cột TotalCharges có khoảng 11 giá trị bị thiếu (NaN). Nguyên nhân do tenure = 0 (khách mới chưa đóng | Thay thế bằng giá trị trung vị (Median) hoặc 0. |

| | | |
|---|---|---|
| | tiền). | |
| **Sai kiểu dữ liệu** | TotalCharges đang ở dạng chuỗi (object). | Ép kiểu về dạng số thực (float). |
| Outliers (Ngoại lai) | MonthlyCharges có phân phối lệch, xuất hiện một số giá trị rất cao. | Kiểm tra và giữ nguyên nếu là giá trị thực tế hợp lý. |
| **Class Imbalance** | Dữ liệu mất cân bằng nghiêm trọng: **No (~73%)** vs **Yes (~27%)**. | Sử dụng kỹ thuật lấy mẫu (SMOTE) hoặc tinh chỉnh trọng số (Class Weight). |

## 2.3. Phân tích khám phá dữ liệu (EDA Highlights)

Các biểu đồ dưới đây minh họa những yếu tố ảnh hưởng mạnh nhất đến quyết định rời bỏ:






- **Tỷ lệ Churn chung:** Dữ liệu cho thấy sự mất cân bằng rõ rệt, nhóm khách hàng trung thành chiếm đa số.
- **Ảnh hưởng của loại hợp đồng (Contract):** Khách hàng sử dụng hợp đồng theo tháng (**Month-to-month**) có tỷ lệ rời bỏ cao đột biến (~45%), trong khi khách hàng ký 1-2 năm rất ổn định.
- **Ảnh hưởng của dịch vụ Internet:** Nhóm sử dụng cáp quang (**Fiber Optic**) có xu hướng rời bỏ cao hơn nhóm dùng DSL, có thể do vấn đề về giá cả hoặc chất lượng dịch vụ chưa tương xứng.

# 3. DATA PREPARATION (TIỀN XỬ LÝ DỮ LIỆU)

## 3.1. Mục tiêu bước tiền xử lý

Bước Data Preparation nhằm đảm bảo dữ liệu đầu vào **sạch, đồng nhất và phù hợp cho mô hình học máy**, tránh lỗi cũng như giảm nhiễu.

Ở bước này, dữ liệu được xử lý theo các nhóm nhiệm vụ chính:

- Xử lý missing values (giá trị bị thiếu)
- Chuyển đổi kiểu dữ liệu (Data Type Conversion)
- Mã hoá biến phân loại (Encoding)
- Chuẩn hoá dữ liệu số (Scaling/Normalization)
- Feature Engineering (tạo thêm thuộc tính hữu ích)

- Tách dữ liệu train/test phục vụ huấn luyện mô hình

### 3.2. Xử lý giá trị thiếu (Missing Values Handling)

Trong dataset, cột **TotalCharges** có một số giá trị bị thiếu (NaN), nguyên nhân thường do khách hàng mới đăng ký dịch vụ chưa phát sinh chi phí.

| Cột | Số lượng missing | Phương pháp xử lý |
|---|---|---|
| TotalCharges | ~10–15 dòng | Chuyển type → numeric và thay bằng median |

📌 Lý do sử dụng **median** thay vì mean là để giảm ảnh hưởng của outlier và phân phối lệch.

### 3.3. Chuyển đổi kiểu dữ liệu (Data Type Correction)

Cột TotalCharges ban đầu ở định dạng **object** (string). Sau khi kiểm tra, cột này được ép kiểu về dạng số:

```
df['TotalCharges'] = pd.to_numeric(df['TotalCharges'], errors='coerce')
```

Giá trị không chuyển được (do chuỗi rỗng) được xử lý tại bước Missing Values phía trên.

### 3.4. Mã hoá dữ liệu phân loại (Encoding)

Dataset chứa nhiều biến dạng phân loại (categorical). Các mô hình học máy không thể xử lý chuỗi nên cần chuyển sang dạng số.

Phương pháp áp dụng:

| Loại biến | Cột liên quan | Kỹ thuật | Lý do |
|---|---|---|---|
| Binary (2 giá trị) | gender, Partner, Dependents, PhoneService, PaperlessBilling, | Label Encoding | Đơn giản, phù hợp logic |

| | | | |
|---|---|---|---|
| | Churn | | Yes/No |
| Multi-class (≥3 nhóm) | InternetService, Contract, PaymentMethod | One-Hot Encoding | Tránh học thứ tự sai của label |

Các cột ID như customerID bị loại bỏ do **không mang thông tin dự đoán**.

### 3.5. Feature Engineering (Tạo thêm thuộc tính)

Để mô hình học tốt hơn, một số thuộc tính mới được tạo ra từ dữ liệu gốc:

| Feature | Công thức | Ý nghĩa |
|---|---|---|
| Revenue | MonthlyCharges × tenure | Tổng doanh thu khách hàng trả |
| AvgMonthlyCharge | TotalCharges / (tenure +1) | Chi phí trung bình hàng tháng |
| LongTermCustomer | tenure > 12 (1=Yes/0=No) | Nhận diện khách hàng trung thành |
| HighCharge | MonthlyCharges > median | Đánh dấu khách hàng chi tiêu cao |

Các đặc trưng này giúp mô hình khám phá sâu hơn hành vi churn.

### 3.6. Chuẩn hoá dữ liệu số (Scaling)

Các thuộc tính số có thang đo khác nhau (tenure, Revenue, MonthlyCharges).
Để tránh mô hình bị **lệch trọng số**, ta sử dụng:

```
scaler = StandardScaler()
```

**Scaling chỉ được fit trên tập Train**, sau đó **transform Test** để tránh *data leakage*.

### 3.7. Tách tập Train/Test

Sau khi tiền xử lý, dataset được chia theo tỷ lệ:

Train: 80%  Test: 20%  Stratify: theo biến Churn để giữ phân bố lớp

Kết quả:

| Tập dữ liệu | Số mẫu | Số thuộc tính |
|---|---|---|
| Train | 5,634 | ~20 features |
| Test | 1,409 | ~20 features |

### 3.8. Kết luận bước Data Preparation

Sau bước tiền xử lý, dataset đã đạt trạng thái:

✓ Không còn missing values

✓ Tất cả biến categorical → numeric

✓ Dữ liệu số được scale đồng nhất

✓ Feature engineering được áp dụng

✓ Train/Test tách đúng chuẩn machine learning workflow

# 4. MODELING (MÔ HÌNH HÓA)

### 4.1. Mục tiêu

Bước Modeling tập trung xây dựng và đánh giá các mô hình dự đoán khả năng rời bỏ của khách hàng (Customer Churn). Các nhiệm vụ chính bao gồm:

- Thiết lập mô hình cơ sở (Baseline model) để làm mốc so sánh.
- Huấn luyện thử nghiệm các thuật toán: Logistic Regression, Random Forest và XGBoost.
- Đánh giá hiệu năng, lựa chọn mô hình tối ưu và lưu trữ (export) dưới dạng file .pkl để phục vụ triển khai.

### 4.2. Chuẩn bị dữ liệu

Nhóm sử dụng tập dữ liệu đã được tiền xử lý và phân chia ở **Bước 3 (Data Preparation)** để đưa vào huấn luyện.

- **Dữ liệu đầu vào:** Tập Train (80%) và Test (20%) đã được mã hóa

(Encoding) và chuẩn hóa (Scaling).

- **Biến mục tiêu (Target):** Churn (0: Không rời bỏ, 1: Rời bỏ).
- **Cơ chế:** Đảm bảo tính nhất quán của dữ liệu để mô hình học được các quy luật chính xác nhất.

### 4.3. Baseline Model (Mô hình cơ sở)

Nhóm chọn Logistic Regression làm baseline vì tính đơn giản, tốc độ xử lý nhanh và khả năng chỉ rõ mức độ ảnh hưởng của từng đặc trưng.

- Kết quả: Độ chính xác (Accuracy) đạt khoảng ~80–82%.
- Nhận xét: Kết quả này là mốc tham chiếu tốt, tuy nhiên mô hình hạn chế trong việc nắm bắt các mối quan hệ phi tuyến tính phức tạp.

### 4.4. Huấn luyện mô hình nâng cao

Để kiểm chứng xem mô hình phức tạp có mang lại hiệu quả tốt hơn mô hình cơ sở hay không, nhóm tiến hành thử nghiệm **Random Forest Classifier**.

- **Đặc điểm:** Là mô hình phi tuyến sử dụng kỹ thuật Bagging với nhiều cây quyết định, thường có khả năng chống overfitting tốt.
- **Kết quả thực nghiệm:** Trên tập dữ liệu này, Random Forest đạt Accuracy khoảng **~79.3%**.
- **Nhận xét:** Mặc dù là thuật toán mạnh, nhưng Random Forest cho kết quả thấp hơn kỳ vọng và thấp hơn so với Logistic Regression. Có thể do dữ liệu của bài toán này phù hợp hơn với các mô hình tuyến tính hoặc mô hình phức tạp đang bị nhiễu bởi một số đặc trưng.

### 4.5. So sánh và Lựa chọn mô hình

Bảng so sánh hiệu quả trên tập Test:

| Mô hình | Accuracy (tập test) | Nhận xét |
|---|---|---|
| Logistic Regression | ~80–82% | Baseline, đơn giản |
| Random Forest | ~79,3% | Hiệu năng thấp hơn ,chưa tối ưu. |

➞ **Quyết định:** Chọn **Logistic Regression** là mô hình cuối cùng vì hiệu năng

cao, ít cần tinh chỉnh phức tạp (tuning) và tránh overfitting tốt nhờ cơ chế lấy mẫu ngẫu nhiên.

### 4.6. Lưu mô hình

Mô hình Logistic Regression sau khi được chọn sẽ được lưu lại bằng thư viện joblib để tích hợp vào ứng dụng dự đoán:

***joblib.dump(best_model, "model.pkl", compress=4)***

### 4.7. Kết luận

Quy trình mô hình hóa đã hoàn tất với việc thử nghiệm từ Logistic Regression đến Random Forest. Mô hình Logistic Regression được xác định là phù hợp nhất và đã được đóng gói thành công (file .pkl) để chuyển sang bước Triển khai (Deployment).

# 5. EVALUATION (ĐÁNH GIÁ MÔ HÌNH)

Sau khi huấn luyện, nhóm tiến hành đánh giá hiệu năng của hai mô hình **Random Forest** và **Logistic Regression** trên tập kiểm thử (Test Set) để tìm ra mô hình tối ưu cho bài toán giữ chân khách hàng.

### 5.1. Bảng so sánh hiệu năng tổng quan

Dựa trên các chỉ số đo lường chính, kết quả so sánh giữa hai mô hình như sau:

| **Chỉ số** | **Random Forest** | **Logistic Regression** | **Nhận xét** |
|---|---|---|---|
| **Accuracy** | ~79.3% | **~81.6%** | Logistic Regression có độ chính xác tổng thể cao hơn. |
| **ROC-AUC** | 0.690 | **0.741** | Logistic Regression phân loại tốt hơn ở các ngưỡng khác nhau. |
| **TP (Churn đúng)** | 175 | **217** | Logistic Regression phát hiện được nhiều khách hàng rời bỏ hơn. |

➞ **Nhận định sơ bộ:** Trong lần thử nghiệm này, mô hình tuyến tính (Logistic Regression) cho kết quả vượt trội hơn mô hình phức tạp (Random Forest), đặc

biệt là khả năng phát hiện đúng nhóm khách hàng rời bỏ.

### 5.2. Phân tích chi tiết (Confusion Matrix & ROC)

Các chỉ số chi tiết được thể hiện qua Ma trận nhầm lẫn (Confusion Matrix) dưới đây:

- **Random Forest Confusion Matrix:**

[[943, 93],
[198, 175]] *(Bỏ sót 198 khách hàng rời bỏ)*

- **Logistic Regression Confusion Matrix:**

[[933, 103],
[156, 217]] *(Chỉ bỏ sót 156 khách hàng rời bỏ)*

**Phân tích ROC Curve:**

- **Random Forest (AUC = 0.690):** Đường cong nằm thấp, khả năng phân tách giữa hai lớp khách hàng chưa thực sự tốt.
- **Logistic Regression (AUC = 0.741):** Đường cong bao phủ diện tích lớn hơn, cho thấy độ tin cậy cao hơn trong dự báo.

### 5.3. Phân tích lỗi (Error Analysis)

Mục tiêu quan trọng nhất của bài toán là **giảm thiểu việc bỏ sót khách hàng sắp rời bỏ (False Negative - Loại II Error)**.

- **Mô hình Random Forest:** Có **198** trường hợp khách hàng thực tế rời bỏ nhưng mô hình dự đoán là "Ở lại". Điều này dẫn đến việc doanh nghiệp bỏ lỡ cơ hội chăm sóc 198 khách hàng này.
- **Mô hình Logistic Regression:** Số lượng bỏ sót giảm xuống còn **156** trường hợp. Đồng thời, mô hình nhận diện đúng **217** trường hợp rời bỏ (so với 175 của RF).

➞ **Kết luận: Logistic Regression là mô hình tốt hơn** để triển khai thực tế trong trường hợp này, vì nó tối đa hóa khả năng giữ chân khách hàng (Recall cao hơn).

### 5.4. Đề xuất chiến lược giữ chân khách hàng (Business Recommendations)

Dựa trên kết quả dự báo từ mô hình, nhóm đề xuất các hành động cụ thể:

1. **Phân nhóm ưu tiên (Priority Targeting):** Tập trung nguồn lực vào danh sách **217 khách hàng** mà mô hình Logistic Regression đã cảnh báo (True Positive). Đây là nhóm có nguy cơ cao nhất cần can thiệp ngay lập tức.
2. **Cá nhân hóa ưu đãi:** Thay vì gửi khuyến mãi đại trà (gây tốn kém), chỉ gửi voucher giảm giá hoặc gói cước ưu đãi cho nhóm được dự báo là Churn (Lớp 1).
3. **Chủ động chăm sóc (Proactive Support):** Với 156 trường hợp mô hình bỏ sót (False Negative), doanh nghiệp cần rà soát lại các dấu hiệu hành vi khác (ví dụ: lịch sử gọi tổng đài phàn nàn) để bổ sung dữ liệu cho các lần huấn luyện mô hình sau, nhằm giảm thiểu tỷ lệ bỏ sót này.

# 6.DEPLOYMENT

**1 Cài đặt dependencies**

```
pip install pandas scikit-learn streamlit joblib
```

**2 Training model**

```
cd src
python modeling.py
```

→ Kết quả: model.pkl và preprocessor.pkl được lưu trong models/

**3 Test predict**

```
python predict.py
```

**4 Chạy demo Streamlit**

```
cd demo
streamlit run app.py
```

→ Mở browser tại: http://localhost:8501

# 7. KẾT LUẬN VÀ KIẾN NGHỊ

## 7.1. Kết luận dự án

Dự án **"Xây dựng mô hình dự đoán Customer Churn"** đã hoàn thành toàn bộ quy trình khai phá dữ liệu theo chuẩn CRISP-DM:

1. **Hiểu dữ liệu:** Đã xác định được các yếu tố chính gây rời bỏ là Hợp đồng theo tháng, Dịch vụ cáp quang và Cước phí cao.
2. **Xử lý dữ liệu:** Đã giải quyết triệt để vấn đề dữ liệu thiếu (Missing values) và mất cân bằng dữ liệu.
3. **Mô hình hóa:** Đã thử nghiệm và so sánh giữa Random Forest và Logistic Regression. Kết quả cho thấy **Logistic Regression** là mô hình phù hợp nhất với bài toán này nhờ tính ổn định và khả năng phát hiện đúng khách hàng rời bỏ (Recall) cao hơn.
4. **Triển khai:** Đã đóng gói thành công mô hình vào ứng dụng thực tế.

**Hạn chế**

- Dữ liệu lịch sử chỉ bao gồm thông tin tĩnh, chưa có dữ liệu hành vi chi tiết theo thời gian thực (ví dụ: tần suất gọi tổng đài phàn nàn trong 7 ngày gần nhất).
- Chỉ số Precision (độ chính xác của các dự báo Churn) còn chưa quá cao, có thể dẫn đến việc khuyến mãi nhầm cho một số khách hàng trung thành (tuy nhiên chi phí này thấp hơn chi phí mất khách).

**7.2. Kiến nghị và Hướng phát triển**

**Về mặt kinh doanh:**

- **Chiến lược hợp đồng:** Cần có chính sách giảm giá sâu để khuyến khích khách hàng chuyển từ hợp đồng "Month-to-month" sang hợp đồng 1-2 năm.
- **Cải thiện dịch vụ:** Tỷ lệ rời bỏ ở nhóm dùng Internet Cáp quang (Fiber Optic) rất cao. Bộ phận kỹ thuật cần kiểm tra lại chất lượng đường truyền và dịch vụ chăm sóc cho nhóm này.

**Về mặt kỹ thuật:**

- Trong tương lai, cần thu thập thêm dữ liệu phản hồi văn bản (Text feedback) của khách hàng để áp dụng các kỹ thuật Xử lý ngôn ngữ tự

nhiên (NLP).

- Thử nghiệm các mô hình Deep Learning (Mạng nơ-ron) khi lượng dữ liệu thu thập được lớn hơn (Big Data).

**Phụ lục**

**Repository : customer-churn-group2-12**

**Link : https://github.com/ngmquann/customer-churn-group2-12**